**Java Fullstack- Angular**

|  |  |
| --- | --- |
| Skill | Interviewed with concepts |
| Coding | Find largest and smallest word in a given sentence. |
| Logical | Given a arithmetic question for quick thinking |
| JavaScript | Var vs let vs const, Hoisting, Arrow fn vs JS fn, Closures, Promises, Async/await, JS programming |
| Angular | Angular Life Cycle, Components and types of components, component and directive, Data sharing b/w components, Dependency injection, services, Observables, RXJS, Routing and Router guards, Angular reactive forms, Change detection strategies, Ways to improve performance |

**Java 8** : Good in java 8 features and multi threaded concepts.Good with collections  
**Springboot**: Basic knowledge in spring boot annotation and features. Good in implementing APIs with JWT for securing the endpoints  
**Rest API**: Have good knowledge in Rest protocals and Http codes and Methods  
**SQL**: Good with sql basic queries  
**Angular**: Have hands on in angular components, data sharing, content projections and template rendering  
**Microservice**: Have good knowledge in microservice design patterns

**Overall** : Good for Full stack

Full Stack (JAVA and React) Developer Job Description

• Should have minimum of 5+ years

• Strong Coding skills and Exposure to Modular Design

• Java/ Core Java • Frames works: Hibernate/Spring/Struts

• Web Services: REST / SOAP

• Spring stack: Spring integration, Jersey, Spring boot, annotations-based Spring config, Spring Data • Micro services - Hands on preferred

• Strong experience - JavaScript, Angular, HTML, CSS

• Experience in UI frameworks like Express, React, Backbone, Angular, etc.

• Good experience in Java and JEE internals (Class loading, Memory Management, Transaction management etc.)

• Cloud Experience would be a plus • Unit test the Developed code and have exposure to End-to-end testing across components and modules Programming language:

• Core Java, OOPS concepts, Collections

• Spring, Spring boot, Hibernate, Micro services,

• Data Structures, Algorithms, Multithreading, Design patterns